Triangle Inequality Variations in the Internet

Cristian Lumezanu, Randy Baden, Neil Spring, Bobby Bhattacharjee
Department of Computer Science
University of Maryland
College Park, MD 20742
{lume, randofu, nspring, bobby}@cs.umd.edu

ABSTRACT

Triangle inequality violations (TIVs) are important for latency sensitive distributed applications. On one hand, they can expose opportunities to improve network routing by finding shorter paths between nodes. On the other hand, TIVs can frustrate network embedding or positioning systems that treat the Internet as a metric space where the triangle inequality holds. Even though triangle inequality violations are both significant and curious, their study has been limited to aggregate data sets that combine measurements taken over long periods of time.

The limitations of these data sets open crucial questions in the design of systems that exploit (or avoid) TIVs: are TIVs stable or transient? Or are they illusions caused by aggregating measurements taken at different times? We collect latency matrices at varying sizes and time granularities and study dynamic properties of triangle inequality violations in the Internet. We show that TIVs are not results of measurement error and that their number varies with time. We examine how latency aggregates of data measured over longer periods of time preserve TIVs. Using medians to compute violations eliminates most of the TIVs that appear sporadically during the measurement but it misses many of the ones that are present for more than five hours.

Categories and Subject Descriptors

C.2.0 [**Computer-communication networks**]: Data communications; C.2.4 [**Computer-communication networks**]: Distributed systems; C.4 [**Performance of systems**]: Measurement techniques; H.4.3 [**Information systems applications**]: Communications applications

General Terms

Measurement, design

Keywords

TIV, triangle inequality violation, latency, variation

IMC'09, November 4–6, 2009, Chicago, Illinois, USA.

1. INTRODUCTION

End-to-end latencies in the Internet demonstrate triangle inequality violations (TIVs). TIVs affect network coordinate [1, 2] and positioning [3] systems and latency-reducing overlays [4]. On one hand, TIVs can be inconvenient for coordinate and positioning systems: because these applications treat the Internet as a metric space—where TIVs are prohibited—inaccurate results may appear. On the other hand, TIVs expose opportunities to improve network routing by offering lower-latency one-hop *detour* [5] paths.

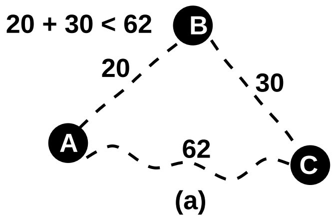
Existing studies on TIVs show that they are widespread and persistent [6, 2, 7]. TIVs are not measurement artifacts, but a natural consequence of the Internet routing [8, 9]. However, all evidence about TIVs has been limited to aggregate latency data sets [2, 10, 11, 6, 3, 8, 7, 9] that combine measurements taken at different times over long periods. These data sets fail to capture the variations of triangle inequalities and may offer false illusions to applications that rely on TIVs or the lack thereof. For example, representing multiple measurements with their median values may reveal TIVs that are short-lived in reality and thus not necessarily a threat for network coordinates, or may miss long-lived TIVs that could be exploited by overlay routing.

The limitations of these data sets open crucial questions for the design of systems that exploit (or filter) TIVs: Are TIVs stable or transient? Are they real or simply illusions caused by aggregating measurements taken at different times? Are they caused by queuing delay or load? And finally, is the performance of these systems affected by the way data is aggregated?

In this paper, we aim to offer new insight into the properties of triangle inequality violations in the Internet, as well as to provide guidelines for better design and evaluation of the systems affected by TIVs. We collect four new latency data sets of different sizes and at varying time granularities. We show that the number of TIVs varies with time and that, when aggregating multiple measurements using medians or minimums, as all previous evaluations have done, we *underestimate* the number of TIVs that existed at any point during the measurement. We propose two additional measurement aggregation techniques and discuss the advantages and disadvantages in applying them in the evaluation of network coordinates and detour routing.

Our contributions can be summarized as follows:

- we collect new data sets, of various sizes and granularities, better suited for analyzing TIVs in a dynamic network environment (§ 3);
- we present a new study on triangle inequality violations in the Internet; we show that TIVs are real and not illusions of measurements (§ 4) and that they vary with time (§ 5);
- we analyze four different methods of computing TIVs from individual measurements and discuss their effects on the per-



time	AB	BC	AC	TIV?
t1	103	40	135	no
t2	90	26	106	no
t3	135	25	139	no
med	103	26	135	yes

(b)

time	AB	BC	AC	TIV?
t1	78	47	140	yes
t2	98	15	135	yes
t3	100	50	166	yes
med	98	47	140	no

(c)

Figure 1: a) Example of triangle inequality violation, b,c) Median values can create the illusion of TIVs. Latencies for AB, BC and AC are measured several times. We show the values at $t1$, $t2$ and $t3$. The final data set is compiled from the medians: although at no time-step is there a TIV among A, B and C, the medians indicate otherwise (b); alternatively, even if each measurement indicates the presence of a TIV, the medians do not reflect it in the final data set (c). All latencies are derived from real measurements.

formance of detour routing and network coordinate applications (§ 6).

2. MOTIVATION

A triangle inequality violation occurs among a triple of nodes in the Internet when the latencies between them cannot form a valid triangle. Figure 1(a) presents such a scenario. We call a triple of nodes that violates the triangle inequality a *bad triangle*. In the bad triangle ABC, AC is the *long side* while AB and BC are the *short sides*. Alternatively, borrowing terminology from Detour [12, 5], we refer to the path (A,B,C) as the *detour path* and to the path (A,C) as the *direct path*.

TIVs are important for latency-sensitive distributed applications such as network coordinate systems or latency-reducing overlay routing. Network coordinates [2, 1] assign positions in a geometric space to Internet hosts, such that the distance between the positions estimates the real latency between hosts. Any three points that form a bad triangle cannot be embedded accurately into a space that prohibits TIVs—such as a geometric space. Thus, the more triangle inequalities there are, the less precise the embedding is [13, 6]. Conversely, that network coordinates do not work well with metric spaces can also be helpful [7]. Embedding errors expose shorter paths between nodes and make them available for overlay routing. Pairs of nodes that are long sides in bad triangles may benefit from detours; pairs that are short sides may be part of shorter detours. These nodes can discover whether they form a long or short side by simply computing the embedding distance to other nodes and comparing it with the real network distance.

Existing evidence about TIVs is derived from aggregate all-to-all latency data sets that combine many measurements [6, 2, 14, 8]. The final latency between two nodes is obtained by taking the median [2] or the minimum [3, 8, 11] of measurements performed over long periods of time such as days or even weeks. Although these data sets are meant to reflect the real Internet latency space, they may fail to accurately depict the characteristics of TIVs. Consider an experiment that measures the latencies among nodes A, B and C at regular intervals and computes the final latency value for each pair as the median of the measured values. In Figure 1, we show values of latencies at three intervals, $t1$, $t2$, and $t3$, as well as the median. These values are derived from real Internet experiments. Although at no time during the measurement was there a triangle inequality violation among A, B and C, the medians indicate otherwise (Figure 1(b)). The opposite can also be true: the triple A, B and C violates the triangle inequality at every time step, but this is not reflected by the medians (Figure 1(c)).

Scenarios such as the ones above reveal the potential pitfalls of reasoning about triangle inequality violations with aggregates of data. Some TIVs may appear when computed with median values for latency but may not be long-lived enough to be significant. Further, aggregates of data may not capture TIVs that, although do not appear continuously during the data collection, may still be present for enough time to be useful for an overlay routing network or to cause embedding errors in coordinates.

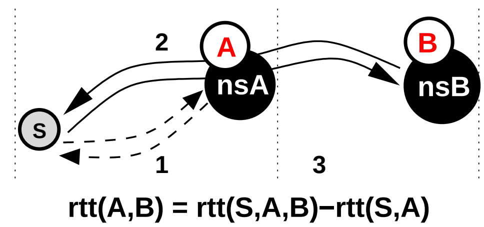


Figure 2: How King works: 1) S measures the RTT to the closest recursive name server of A, nsA, 2) S sends a recursive query through nsA for a domain resolved by a name server of B, nsB and measures its round-trip time, 3) the latency between A and B is estimated as the difference between the time taken to perform the previous two operations.

Data set	Nodes (Pairs)	Duration	Interval
K200-1000pairs-5min	200 (1000)	24h	5min
K200-allpairs-1h	200 (all)	44h	1h
K200-allpairs-3h	200 (all)	30h	3h
K1715-allpairs-2d	1715 (all)	20d	2d

Table 1: Latency data sets. For each set we show: a) the name, b) the total number of nodes (and the number of pairs measured), c) the duration of the experiment, and d) the average interval between consecutive measurements of the same pair. All data sets were collected in the period March-April 2008.

3. METHODOLOGY

We use the King tool [15] to collect latency data sets that are better suited for studying triangle inequality violations. King is the only tool that estimates all-to-all round-trip times between **any** hosts in the Internet.

3.1 King

King uses recursive DNS queries to estimate the latency between two hosts in the Internet. Given the IP addresses of two nodes, King computes the propagation delay between them as the delay between authoritative name servers for those addresses. Figure 2 shows an example. A user located at S tries to estimate the latency between hosts A and B. First, S measures the round-trip time to

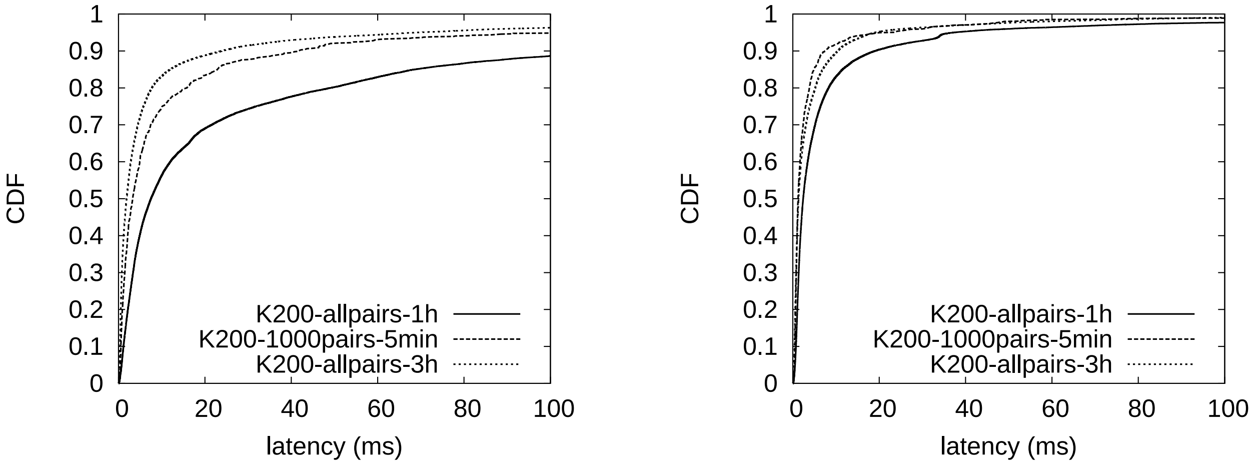


Figure 3: Cumulative distributions for left) Standard deviation and right) Interquartile range for three data sets

nsA, the closest recursive name server of A. In our measurements, to minimize the error of estimation, we ensure that A and nsA are in the same subnet. Then, S asks nsA to recursively resolve a name served by a name server of B, nsB. The latency between nsA and nsB is obtained by subtracting the times taken to perform the two operations and represents an estimate of the latency between hosts A and B.

King is the only tool that estimates all-to-all round-trip times between any hosts in the Internet. Although King latencies have been criticized for not being representative for latencies between end hosts, our goal is not to verify these claims. For the purposes of analyzing TIVs, we believe that King's advantages outweigh its shortcomings.

3.2 Data Sets

We collect four latency data sets of various sizes and at different time granularities. The IP addresses of the nodes in our measurements are of users participating in a file sharing application and are available through the Vivaldi project [2]. The chosen IPs share the same subnet with their authoritative name servers so that better-connected DNS servers would not influence the estimates of inter-client latencies.

We describe the properties of the data sets in Table 1. Our goals are to collect data sets that are synchronous: all pairs of nodes are measured at least once within a predefined time interval. The size of the interval determines the granularity of the data set. We use four sampling intervals: 5 minutes, 1 hour, 3 hours and 2 days. At the beginning of each interval, we run King for all pairs of nodes in the data set from a computer at the University of Maryland. Each individual King measurement consists of four consecutive probes, out of which we keep the minimum value. Collecting latencies at smaller time granularities provides more accurate snapshots of the latency space. However, it also limits the number of pairs that we can measure accurately, without unnecessarily loading the DNS servers or the source computer. Thus, for the smaller granularities, we limit the scope of the measurement to 200 IP addresses (1000 pairs chosen at random for the 5 minute interval and all pairs for the 1 hour and 3 hour intervals). We collect a much larger data set (1715 IP addresses) when the granularity is increased to two days. Because we want to capture the dynamic properties of TIVs, we present results only for the three data sets with finer granularity.

4. LATENCY VARIABILITY

Latency variation on a path may lead to TIVs; conversely, if we perceive latencies to be varying (when the underlying path is stable), we may assert the existence of fake TIVs. In the rest of this section, we classify the causes of the recorded latency variations in our measurements. We show that the chances of inferring fake TIVs is small, and that most latency variation can be attributed to changes in load or changes in routing.

4.1 Measurements Vary Over Time

We study how end-to-end round-trip time varies for the duration of the measurement. We use two measures of variability: standard deviation (STD) and interquartile range (IQR). Standard deviation represents the variability of all data points equally, while interquartile range—the difference between the 75th and 25th percentiles—measures the variability of the 50% of points around the median. Figure 3 shows the cumulative distributions of STD and IQR for the three data sets with small granularities. Each point on the plots is associated with a pair of nodes. We make the following observations:

- All distributions have long tails; each data set has a few pairs of nodes that exhibit high variations in latency. 5% of the pairs in K200-1000pairs-5min and K200-allpairs-3h and 12% of the pairs in K200-allpairs-1h have standard deviations of more than 100ms.
- Second, in all data sets, less than 10% of the pairs have interquartile ranges of more than 40ms. Combined with the previous observation, this implies that the variability of the latency comes mainly from the more extreme values, rather than values closer to the median.
- Finally, the pairs in K200-allpairs-1h have higher standard deviations than the pairs in K200-allpairs-3h. This suggests that variability decreases with an increase in sampling interval. We confirm that this is true in Section 4.2.

4.2 Causes of Variations

Determining the exact cause that leads to each latency change is difficult. Instead, we classify the possible causes of variation into three categories:

- Load-based causes refer to events such as queuing delay at the routers or transient load at the DNS servers involved in

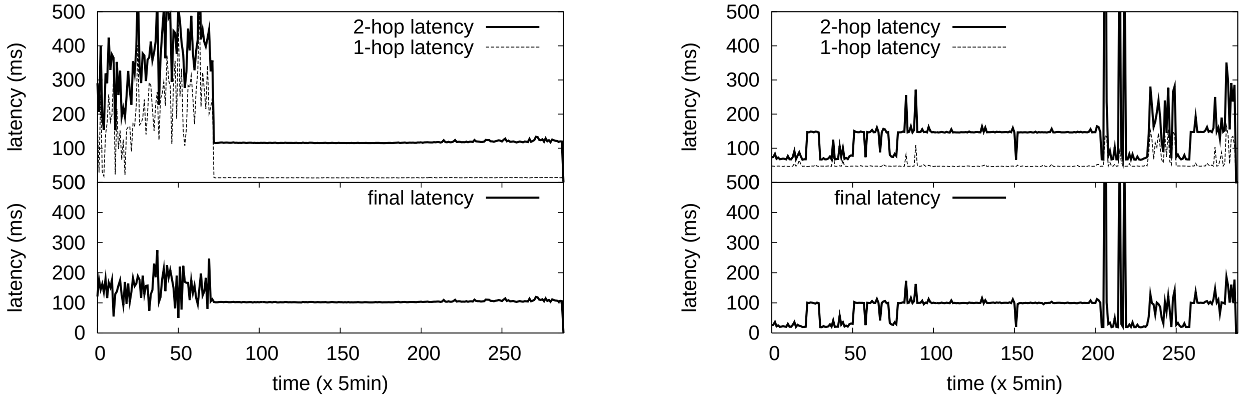


Figure 4: Examples of latency variations between pairs of nodes: left) the latency between 66.189.0.29 and 200.31.70.18 exhibits variations due to load; because both 1-hop and 2-hop latencies have similar variations, we conclude that it is either network load from S to A or DNS load on A; right) the latency between 216.61.143.252 and 147.136.250.51 varies during the duration of the measurement; besides the occasional spikes given by load, there are long periods of time (from 1h to 8h) when the latency changes significantly (by 70ms)

measurements. They are likely to manifest as short-duration spikes or oscillations [16].

- Routing-based causes are path changes in the Internet determined by link or node failures or by routing changes. Although routes can also oscillate, their oscillations tend to have longer durations [17]. Thus, path changes are more likely to trigger longer-term changes in latencies.
- Measurement-based causes depend on the parameters of the measurement process. We consider two potential sources of variation: the sampling interval and the time at which we measure each sample. Since we limited the number of pairs probed per sampling interval to avoid unnecessarily loading DNS servers, we do not consider load on name servers a measurement-based cause of variation.

Routing-based and Load-based causes.

We focus first on the routing-based and load-based causes of variation. We select two pairs from the K200-1000pairs-5min data set and show their latency distributions in Figure 4. We define the latency from the source of the measurement to the first DNS server (nsA in Figure 2) as the 1-hop latency, and the latency from the source to nsB through nsA as the 2-hop latency. The final latency is obtained by subtracting the two values. We show the distributions of 1-hop and 2-hop latencies in the top part of Figure 4. Every point on the plot is associated with one measurement. We make the following observations:

- the variation of latency in Figure 4(left) exhibits many short-duration oscillations for the first 350 minutes; this is most likely a load-based event. After 350 minutes, the latency stops oscillating.
- the variation of latency in Figure 4(right) shows fewer oscillations and the latency tends to stabilize around two values (30ms and 100ms) for periods ranging from 1 hour to 12 hours; this behavior suggests a routing-based event.
- in Figure 4(left), the variations of the final, 1-hop and 2-hop latencies follow the same trends; in Figure 4(right), the 1-hop latencies remain constant over the first 200 intervals, while the 2-hop latencies change; this indicates the location of the event that causes the variation: a spike that appears on the 2-hop latency distribution but not on the 1-hop latency distribution must be caused by an event that occurred on the path between the two DNS servers.

We compute the sample correlation for the 1-hop and 2-hop latencies for each pair in K200-allpairs-1h. We use this data set because it has the smallest time granularity of the ones that contain all-pair latencies and because it exhibits the greatest variability. Figure 5(left) shows that there is less correlation among the pairs with the top 5% interquartile ranges—these are the more variable pairs. This indicates that the source of high variance typically lies between the two DNS servers probed by King. It also suggests that the source of the measurement has less impact on the variability of the data, as we show in Section 4.2.

We also compute the average absolute difference between consecutive measurements for each pair in K200-allpairs-1h. The average consecutive difference estimates how a prior measurement predicts a future one. A low average consecutive difference indicates that the data varies with low frequency (and the variation is likely due to a routing-based cause), while a high average consecutive difference indicates that the data varies with high frequency (possibly due to a load-based cause). Figure 5(right) shows that, for the pairs in the top 5% among interquartile ranges, the average consecutive difference is larger than for pairs in general. Indeed, we would expect that the more variable pairs have higher average consecutive differences. Less than 20% of those high-variance pairs have at most 30 ms of average consecutive differences; in those cases the cause of the variance is most likely due to path changes. For the remaining pairs, the variance changes rapidly; the source of the variance in those cases is most likely due to loaded DNS servers or high queuing delay at routers.

Measurement-based causes of variation.

Another source of variation may be the process of measurement itself. Next we verify whether the sampling interval or the time when we measure each sample affect the variability of the data.

We split the K200-1000pairs-5min data set into k more coarsely grained subsets. In each subset, measurements for the same pair of

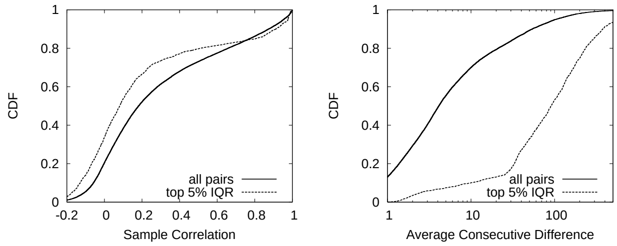


Figure 5: Cumulative distributions of left) sample correlation between 1-hop and 2-hop latencies, and right) average difference between consecutive latency measurements, in K200-allpairs-1h, for all pairs and for the top 5% of pairs when ordered by IQR.

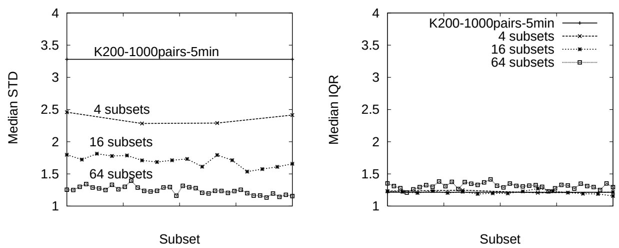


Figure 6: Median left) standard deviations and right) interquartile ranges for the pairs in each subset in the K200-1000pairs-5min data set. Each point represents the median value for one of the subsets. As the sampling interval decreases, so does the median standard deviation.

nodes are collected at $k \times 5$ minute intervals [1]. For example, when using $k = 4$ subsets, subset i contains all measurements taken at sample intervals i, $i + 4$, $i + 8$ and so on. Dividing the original data set in this way allows us to obtain k different measurement sets with $k \times 5$ minute sampling intervals. All subsets appear to start at five minute intervals over the course of $k \times 5$ minutes.

We compute the standard deviation (STD) and the interquartile range (IQR) for all pairs of nodes in each of the subsets, for $k = 1$ (the entire data set), $k = 4$, $k = 16$, and $k = 64$. Figure 6 shows the median STD and IQR. While the median STD decreases when using sparser samples, the median IQR remains approximately the same. We would expect that by sampling less often we are less likely to measure unusually high values. However, such values are always above the 75th percentile of the data, so they do not significantly affect the IQR. Also, the median STD and IQR do not change significantly between subsets, indicating that the time at which the measurement starts does not affect latency variance.

This analysis highlights a trade-off between sampling rate and the effect of rare high-latency measurements when we use a fixed list of the most recent latency measurements. A high-frequency sampling rate will observe more high-latency measurements, but those measurements will pass through the list more quickly. A low-frequency sampling rate does not observe high-latency measurements very often, but when it does, they remain in the list for a long period of time. This variation is typically mitigated through the use of median latency measurements, but since the IQR remains relatively stable it would also be safe to consider alternative aggregates (such as the mean) restricted to the middle 50% of the data.

In conclusion, 10% of the pairs of nodes in our data sets exhibit significant changes in latency for the duration of the measurement.

[1] We use 5 minute intervals because this is the granularity of K200-1000pairs-5min (see Section 3)

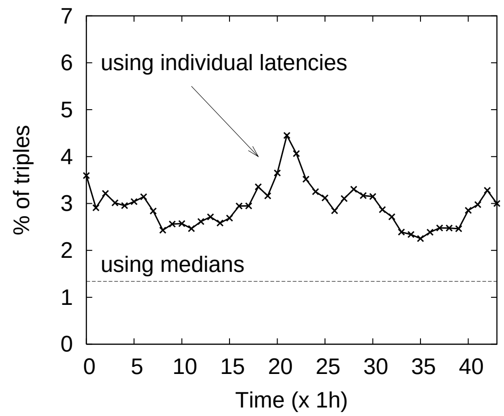


Figure 7: Percentage of TIVs out of the total number of triangles for the K200-allpairs-1h data set.

These variations are due mainly to load-based and routing-based causes and do not come from non-optimal choices of measurement parameters such as sampling interval or start time. Thus, we ensure that the triangle inequality violations we study next are not measurement illusions but real properties of the latencies we collected.

5. TRIANGLE INEQUALITY VARIATIONS

In this section, we study the variation of triangle inequality violations and examine how well aggregate data sets that combine measurements taken over long periods of time capture the TIVs that were present during the measurements.

5.1 TIVs vary over time

We count the number of triangle inequality violations after each sampling interval in the K200-allpairs-1h data set. We consider only those violations for which the difference between the sum of the short sides (the detour path) and the long side (the direct path) is larger than both 10ms and 10% of the latency of the long side. By considering only those violations that are significant, we protect our results from overstating the number of TIVs because of measurement error. Furthermore, applications that use triangle inequality violations to identify detour paths seek significant violations due to the overhead of relaying along the detour path.

Figure 7 shows the number of TIVs at every hour during the measurement. The vertical axis represents the percentage of bad triangles after each interval, out of all triples that have been measured during the interval. We define the median TIVs to be the TIVs computed using the median latency for each pair. The percentage of median TIVs is represented by the horizontal line at 1.34%. Figure 7 indicates that triangle inequality violations vary in time. However, at no point during the measurement process is the number of violations lower than what we would obtain using the medians. Thus, data sets that represent multiple measurements by their median values are conservative: they reveal fewer triangle inequalities than there were during the measurement process. Of course, if the lost TIVs are all short-lived, it may be beneficial not to reveal them; for instance, we only want to use long-lived TIVs for finding detour paths. We study next the longevity of TIVs.

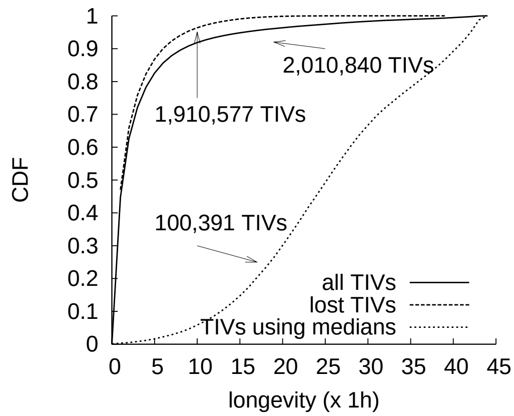


Figure 8: Cumulative distribution of the longevity of TIVs in the K200-allpairs-1h data set.

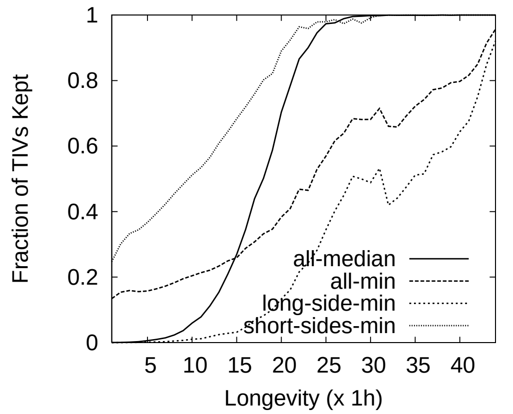


Figure 9: Probability distribution of the fraction of TIVs that appear during the measurement and are preserved when computed on the aggregate data using one of the four methods: all-median, all-min, short-sides-min, long-side min.

5.2 Longevity

What happens to a TIV seen at one point during the measurement? Does it appear in the set of TIVs computed with medians? We expect that, due to extreme values in latency measurements, many triangles are short-lived—they are the effect of an unusually high latency.

We define the longevity of a TIV as the number of intervals in which it appears. We do not require the intervals to be consecutive to avoid bias due to missing or extreme measurements. We compute the longevity for three categories of TIVs: all TIVs seen during the measurement (intermediate TIVs), all median TIVs, and all TIVs seen during the measurement but not when using medians (lost TIVs). Figure 8 shows the distributions of longevities for TIVs in the three categories.

More than 80% of all TIVs have a longevity of less than 5 hours, while almost all ($\geq$99%) TIVs computed with medians are seen for more than 5 hours and more than half of them for more than a day. Thus, using medians eliminates the short-term TIVs.

Of all TIVs, only 18% have a longevity of more than 5 hours. However, of these long-lived TIVs, 72% (not shown in the figure) are lost—they do not appear as median TIVs. Such violations are present long enough to be able to help an overlay routing application—by exposing a shorter detour—but are not captured when the measurements are aggregated.

Scenarios where the medians create a TIV that does not exist, as in Figure 1, are extremely infrequent. For example, 128 triangles (0.1%) appear only when using medians and never during the measurement. Using medians only ignores 1.5% of the TIVs that appear more than half the time using individual latency measurements.

6. ALTERNATIVE WAYS TO COMPUTE TIVS

In this section, we propose alternative ways to compute TIVs from intermediate measurements and discuss their effects on the performance of latency-sensitive distributed applications. Network coordinate and positioning systems do not adapt to TIVs very well: preserving many intermediate TIVs in the final data set will likely provide a lower-bound on performance. On the other hand, detour routing applications perform better when more TIVs are available.

We investigate four ways of computing the number of TIVs: *all-median*, *short-sides-min*, *long-side-min* and *all-min*. We described *all-median* in Section 5.1. In *short-sides-min*, when we verify whether a triple forms a TIV, we consider the minimum latencies for the potential short sides and the median latency for the long side. In *long-side-min*, we use medians for the short sides and minimum for the long side. In *all-min*, as in other previous studies [11, 3], we use the minimum latency values for every edge of the triangle.

All-median is conservative. While it eliminates many short-term TIVs, it also ignores 72% of the TIVs longer than five hours that appear during the measurement (§ 5.2). Intuitively, the *long-side-min* method decreases the number of TIVs that are preserved and provides a more conservative data set for evaluating latency-reducing overlay networks. On the other hand, the *short-side-min* approach preserves more TIVs and offers a worst case scenario for network coordinates.

We define final TIVs as the violations computed using aggregates over intermediate measurements, while the kept (or preserved TIVs) are the intermediate TIVs that are also final. Figure 9 shows the distribution of the fraction of TIVs that are kept by each of the four methods. Every point represents the fraction of TIVs that are kept for each longevity value. Table 2 shows the percentage of intermediate TIVs that are preserved and of final TIVs that never appear during the measurement (are not intermediate TIVs).

Short-sides-min loses less long-lived TIVs than *all-median* but also keeps more short-lived TIVs. Of all TIVs longer than 5 hours, *all-median* keeps 28% while *short-sides-min* keeps almost 60%. Using either of the two methods will better reflect the performance of latency-reducing detour routing applications. *Short-sides-min* keeps more TIVs but also keeps more than 15% of triples which never violate the triangle inequality in individual measurements. *All-median* provides a more conservative estimation, biased towards keeping long TIVs and losing short ones.

All-min and *long-side-min* keep about as many short-lived TIVs as *short-sides-min* and *all-median* respectively. However, neither *all-min* nor *long-side-min* keep as many of the very long-lived TIVs as the other two methods. In conclusion, the *short-side-min* method of computing TIVs is suitable for applications that require an upper-bound on the number of TIVs. It helps provide a lower bound on the performance of network coordinate systems. Although used in the evaluations of several network coordinate and positioning systems [3, 11], *all-min* understates heavily the number of TIVs (it keeps only 5%) and thus does not provide an accurate latency snapshot for evaluation.

Method	Intermediate TIVs preserved	Intermediate TIVs w long ≤ 5 preserved	Intermediate TIVs w long > 5 preserved	Final TIVs that are false
all-median	4.9%	0.1%	28.1%	0.006%
all-min	23%	21.6%	30%	6%
short-sides-min	49.1%	46.6%	60.8%	15.3%
long-side-min	1.9%	0.01%	11%	< 0.001%

Table 2: Percentage of TIVs preserved or added by the various methods out of the total number of TIVs in the corresponding categories. For instance, out of all intermediate TIVs, we preserve 49.1% with the short-sides-min method. 15.3% of the TIVs computed with short-sides-min do not appear at all during the measurement.

7. RELATED WORK

We divide previous research related to triangle inequality violations in the Internet into two parts: studies on end-to-end latency [12, 8, 4, 9] and studies on the performance of network coordinate systems [13, 6, 18].

Savage *et al.* [12] measure latencies between geographically diverse Internet nodes and show that more than 20% of the pairs of nodes form long sides in TIVs. Zheng *et al.* [8] argue, using data collected from the GREN research network, that TIVs are a persistent, widespread and natural consequence of Internet routing policies. These studies are limited to aggregate data sets computed over long periods of time. None of them treat TIVs as dynamic properties of the Internet. We use real-world latency data sets to show that the number of TIVs varies with time and that by aggregating data with medians or minimums of many measurements, we risk missing many existing violations.

Several studies examine TIVs in relation to the impact they have on network coordinate [2, 1] and positioning systems [3]. Because these systems treat the Internet as a metric space—where TIVs are prohibited—they may obtain inaccurate results. Lee *et al.* [13] show how TIVs in latency data sets affect the accuracy and suitability of embeddings. Wang *et al.* [6] identify problems caused by TIVs in the neighbor selection process of embedding and positioning algorithms [2, 3] and propose a simple TIV alert that eliminates the measurements that lead to severe violations. That triangle inequality violations frustrate network coordinates is not necessarily bad. PeerWise [4] uses embedding errors in coordinate systems to discover which pairs of nodes are more likely to benefit from a detour (*i.e.*, are long sides in TIVs) or offer a detour (*i.e.*, are short sides in TIVs). All of these studies treat TIVs as a static network property and compute violations based on combinations of multiple measurements. They may benefit from our observation that we can conservatively estimate the number of TIVs with the minimum instead of the median for the short sides of the triangles.

8. CONCLUSIONS

In this paper, we offer new evidence into the properties of Internet triangle inequality violations. We show, using real world latency data sets of varying sizes and granularities, that TIVs are real and not merely illusions or artifacts of measurements. The number of TIVs varies over time and the TIVs present during the measurement are not necessarily preserved when many measurements are aggregated using median or minimum latencies. We provide simple guidelines for the evaluation and design of systems whose performance depends on triangle inequality violations, such as network coordinates or detour routing.

9. REFERENCES

[1] T. S. E. Ng and H. Zhang, "Predicting Internet network distance with coordinates-based approaches," in *INFOCOM*, 2002.

[2] F. Dabek, R. Cox, F. Kaashoek, and R. Morris, "Vivaldi: a decentralized network coordinate system," in *SIGCOMM*, 2004.

[3] B. Wong, A. Slivkins, and E. G. Sirer, "Meridian: A lightweight network location service without virtual coordinates," in *SIGCOMM*, 2005.

[4] C. Lumezanu, R. Baden, D. Levin, N. Spring, and B. Bhattacharjee, "Symbiotic relationships in Internet routing overlays," in *NSDI*, 2009.

[5] S. Savage, T. Anderson, A. Aggarwal, D. Becker, N. Cardwell, A. Collins, E. Hoffman, J. Snell, A. Vahdat, G. Voelker, and J. Zahorjan, "Detour: A case for informed Internet routing and transport," *IEEE Micro*, vol. 19, no. 1, pp. 50–59, 1999.

[6] G. Wang, B. Zhang, and T. S. E. Ng, "Towards network triangle inequality violation aware distributed systems," in *IMC*, 2007.

[7] C. Lumezanu, D. Levin, and N. Spring, "PeerWise discovery and negotiation of faster paths," in *HotNets*, 2007.

[8] H. Zheng, E. K. Lua, M. Pias, and T. G. Griffin, "Internet routing policies and round-trip times," in *Passive and Active Measurement Workshop*, 2005.

[9] C. Lumezanu, R. Baden, N. Spring, and B. Bhattacharjee, "Triangle inequality and routing policy violations in the Internet," in *PAM*, 2009.

[10] J. Stribling, "Planetlab all pairs ping," http://www.pdos.lcs.mit.edu/~strib/pl_app/.

[11] B. Zhang, T. E. Ng, A. Nandi, R. RIedi, P. Druschel, and G. Wang, "Measurement-based analysis, modeling, and synthesis of the internet delay space," in *IMC*, 2006.

[12] S. Savage, A. Collins, E. Hoffman, J. Snell, and T. Anderson, "The end-to-end effects of Internet path selection," in *SIGCOMM*, 1999.

[13] S. Lee, Z.-L. Zhang, S. Sahu, and D. Saha, "On suitability of euclidean embedding of internet hosts," in *Sigmetrics*, 2006.

[14] J. Ledlie, P. Gardner, and M. Seltzer, "Network coordinates in the wild," in *USENIX NSDI*, 2007.

[15] K. Gummadi, S. Saroiu, and S. Gribble, "King: Estimating latency between arbitrary Internet end hosts," in *IMW*, 2002.

[16] V. Paxson, "End-to-end Internet packet dynamics," in *ACM Sigcomm*, 1997.

[17] C. Labovitz, A. Ahuja, A. Abose, and F. Jahanian, "Delayed Internet routing convergence," in *SIGCOMM*, 2000.

[18] E. K. Lua, T. Griffin, M. Pias, H. Zheng, and J. Crowcroft, "On the accuracy of the embeddings for Internet coordinate systems," in *IMC*, 2005.